McMillan's Complete Prefix for Contextual Nets

Paolo Baldan - University of Padova
Andrea Corradini - University of Pisa
Barbara Koenig - University of Duisburg
Stefan Schwoon - TU Munich

Why?

- Paradigmatic and simple formalism with concurrent read-access to shared resources
- Interest in
  - graph transformation systems (GTS)
  - more general rewriting systems (e.g., on adhesive categories [LS'04])

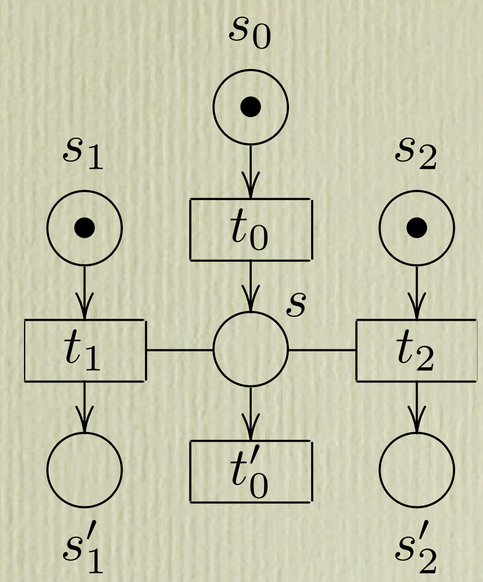
Graph transformation systems

- Models of concurrent and distributed systems
  - graph = state
  - rewriting rules = dynamics
- Some work on unfolding-based verification of GTSs (CN as a conceptual reference)
  - infinite-state systems (approx. of the unfolding)
  - finite-state systems

Outline

- **Contextual nets** and their **unfolding**
- **Complete finite prefix**
  - **encoding** CN as ordinary nets
  - **directly**
    - a non constructive definition
    - a constructive definition and an algorithm

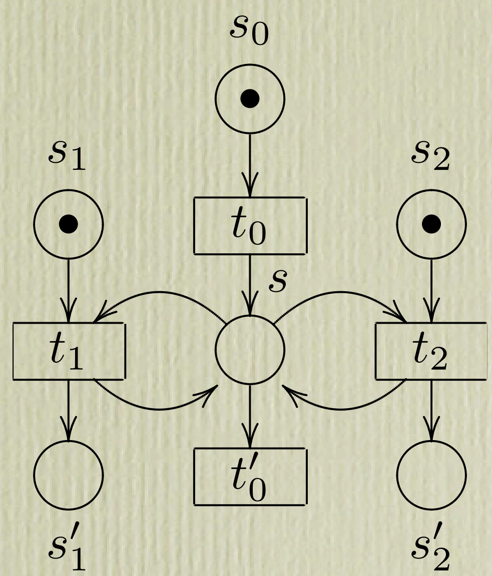
Contextual nets



read arcs
activator arcs
...

- Modelling of concurrent read-access to resources
  - concurrent accesses to shared data, concurrent constraint programs, asynchronous circuits, ...

Encoding as ordinary nets

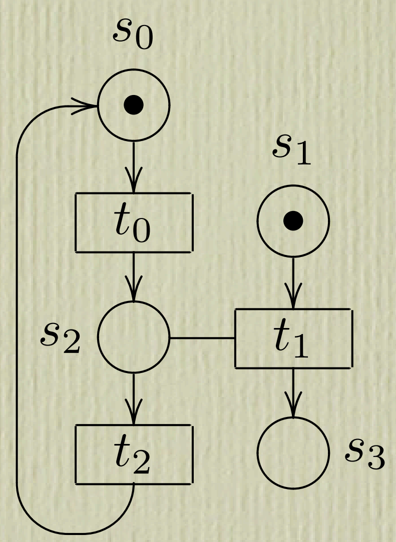


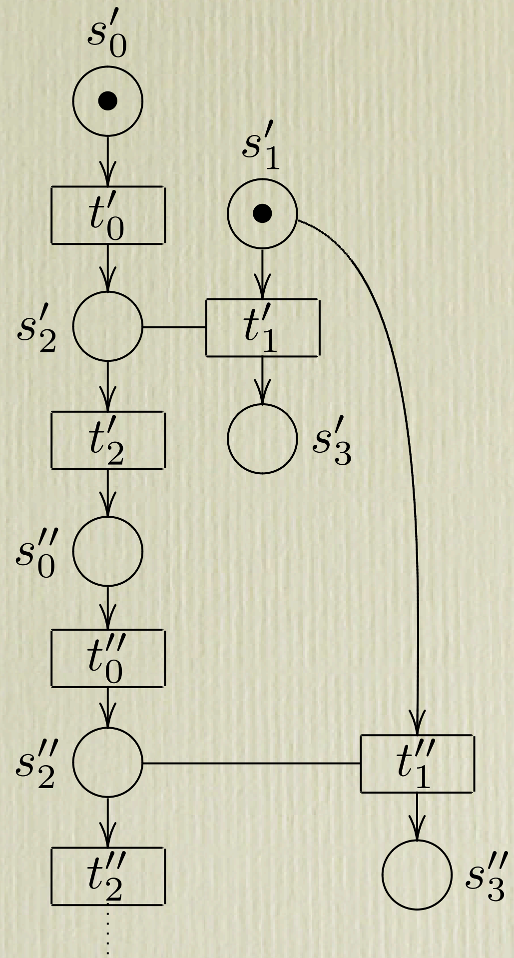
- Fine for reachability
- Loss of concurrency (later a better encoding ...)

Unfolding

- If we unfold the encoding ....
- the number of places and transition may explode due to the forced sequentialization

Unfolding of Contextual Nets





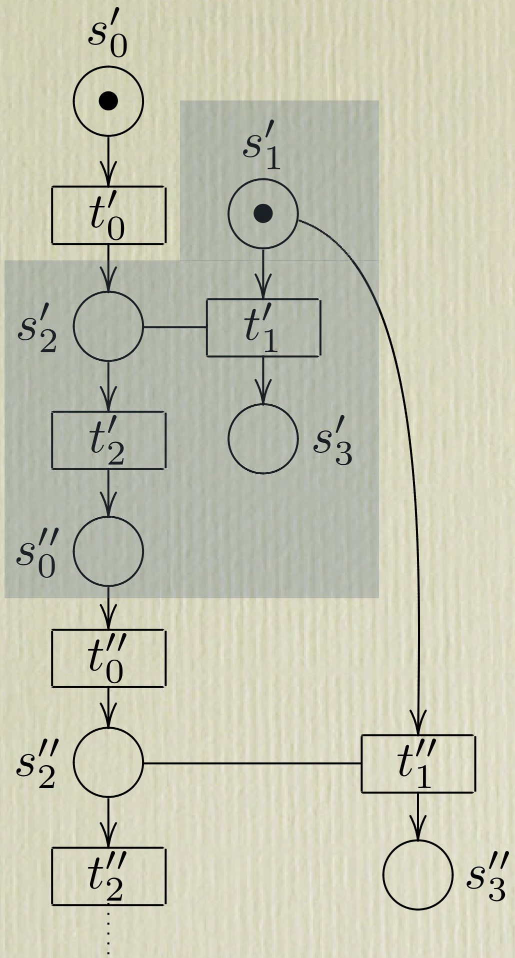
[BCM'98]
[VSY' 98]

Unfolding of Contextual Nets

- New kind of dependency: **asymmetric conflict**

$$t'_1 \nearrow t'_2$$

Prime Event Structures
↓
Asymmetric Event Structures



Generalizing Winskel's Semantics

- From

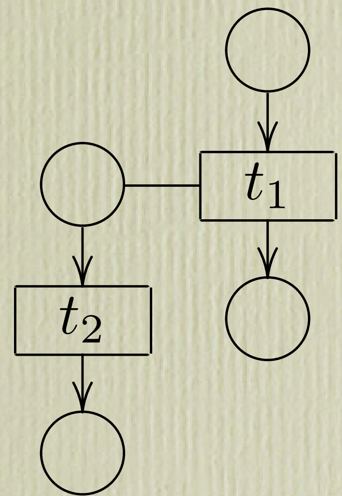
$$\mathbf{Safe} \; \underset{\mathcal{U}}{\overset{\hookleftarrow}{\underset{\perp}{\rightleftarrows}}} \; \mathbf{Occ} \; \underset{\mathcal{E}}{\overset{\mathcal{N}}{\underset{\perp}{\rightleftarrows}}} \; \mathbf{PES} \; \underset{\mathcal{L}}{\overset{\mathcal{P}}{\underset{\sim}{\rightleftarrows}}} \; \mathbf{Dom}$$

- ... to

$$\mathbf{CN} \; \underset{\mathcal{U}}{\overset{\hookleftarrow}{\underset{\perp}{\rightleftarrows}}} \; \mathbf{OccCN} \; \underset{\mathcal{E}}{\overset{\mathcal{N}}{\underset{\perp}{\rightleftarrows}}} \; \mathbf{AES} \; \underset{\mathcal{L}}{\overset{\mathcal{P}}{\underset{\perp}{\rightleftarrows}}} \; \mathbf{Dom}$$

Finite complete prefix for CN

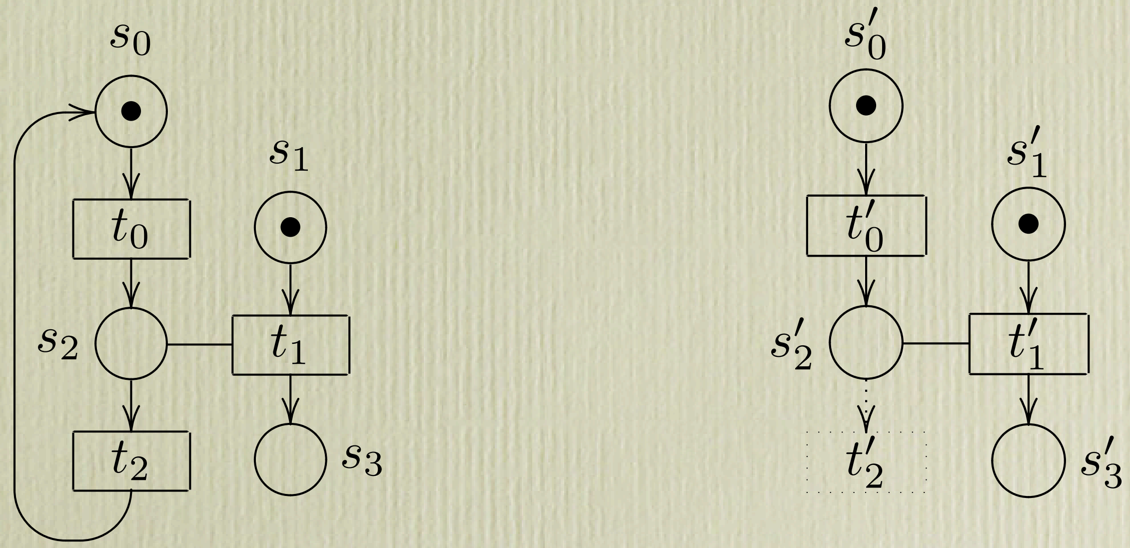
- The construction for ordinary nets does not extend straightforwardly to CN
- [VSY'98] shows that this works for read persistent CN



never enabled at the same time

Finite complete prefix for CN

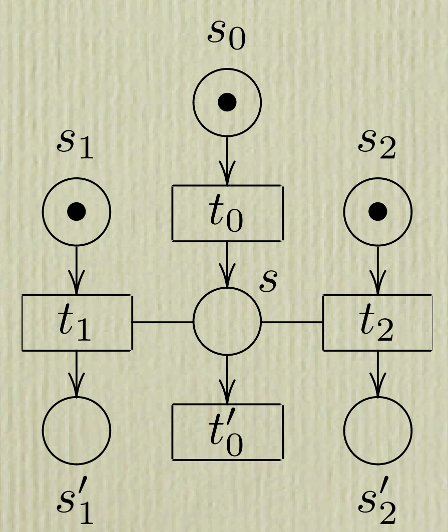
- The construction does not work in general [VSY'98]

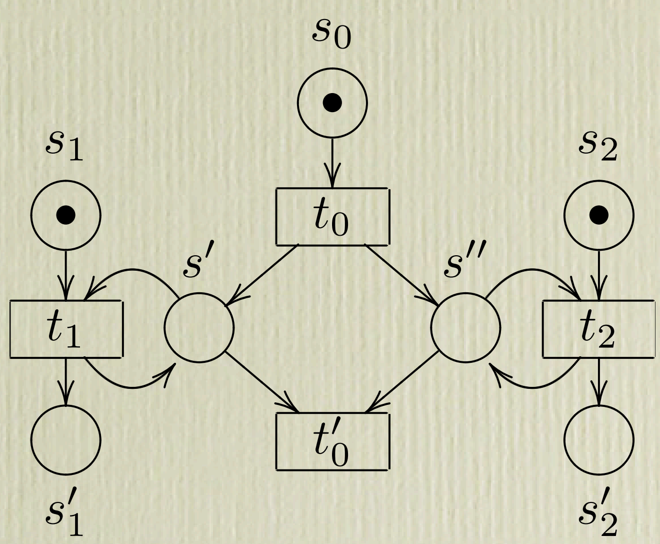


prefix is **not complete**!
(s_0s_3 missing)

PR-encoding for general CN

- **Idea**: create a “private” copy of shared places for each reader [VSY’98]





PR-encoding (cont.)

- The encoding preserves concurrency, but ...
  - the unfolding is an ordinary net (larger)
  - works less fine for non-safe nets
  - Petri-net specific: seems difficult to extend to other formalisms like graph grammars

A direct approach

Problem

- Standard notion:
  - An event is **cut-off** if there is an event with smaller **causal history** producing the same marking

Problem: Causal history

- **Causal history** of an event t in a configuration = set of events which must precede t
  - For ordinary nets

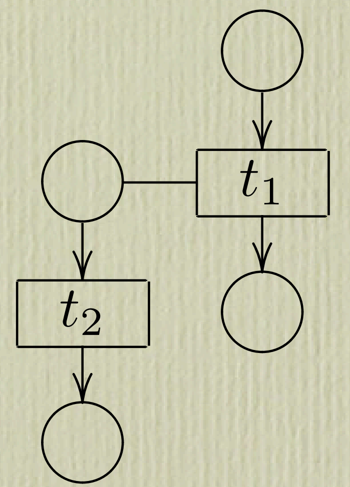
$$C[t] = \{t' \in T \mid t' \leq t\} = \lfloor t \rfloor$$

  - For contextual nets

$$C[t] = \{t' \in C \mid t' \nearrow^* t\}$$

an event can various causal histories!

Example



$$Hist(t_2) = \{\{t_2\}, \{t_1, t_2\}\}$$

First solution

- **Generalised cut-off** t

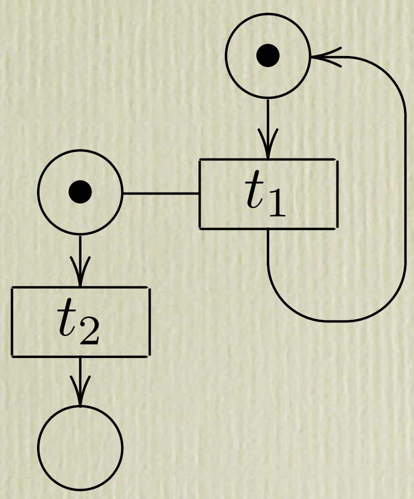
$$\forall H \in Hist(t).\ \exists t'.\ \exists H' \in Hist(t') \text{ s.t.}$$

$$|H'| < |H| \quad \text{and} \quad mark(H) = mark(H')$$

- **Theorem**: The largest prefix without cut-offs is complete (and finite for finite-state nets)

First solution

- **Non constructive**:
  - a transition can have infinitely many histories!
  - cfr. [Winkowski'04]

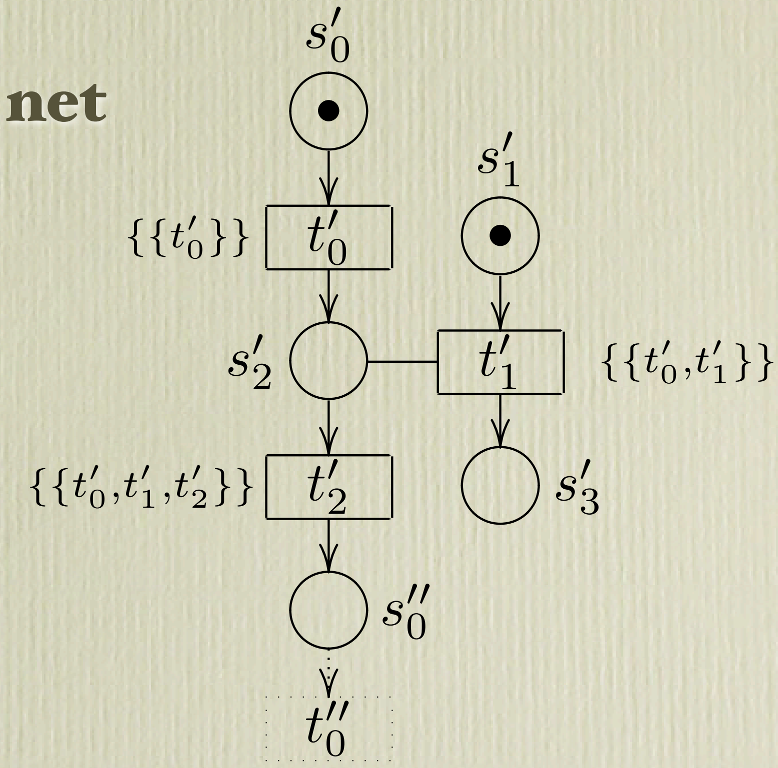


Second solution

- **Idea**: associate to each event a subset of "interesting" histories which contribute to generating new markings
- **Enriched occurrence net**

$$E = \langle N, \chi \rangle$$

$$\emptyset \neq \chi(t) \subseteq Hist(t).$$



Defining a complete prefix

- **Prefix ordering** on enriched occurrence nets

$\langle N_1, \chi_1 \rangle \sqsubseteq \langle N_2, \chi_2 \rangle$ if
- N_1 is a prefix of N_2
- $\chi_1(t) \subseteq \chi_2(t)$ for any t

- **Cut-off**

enriched event $\langle t, H_t \rangle$ s.t. there exists $\langle t', H_{t'} \rangle$

- $mark(H_t) = mark(H_{t'})$ and
- $|H_{t'}| < |H_t|$.

Defining a complete prefix

- **Truncation**: largest enriched prefix of the unfolding without cut-offs
- **Theorem**: The truncation is complete (and finite for finite-state nets)

Algorithm

- **Idea**: construct incrementally a finite closed prefix of the unfolding, avoiding to introduce useless histories

Algorithm

- Refers to
  - **Fin** = current prefix (enriched occ. net)
  - **PE** = possible extensions <t, H_t> where
    - t is an event enabled in *Fin* and
    - H_t a history of t in *Fin*

Algorithm

Fin = initial marking

$\mathsf{PE} = pe(\mathsf{Fin})$

while($\mathsf{PE} \neq \emptyset$) **do**

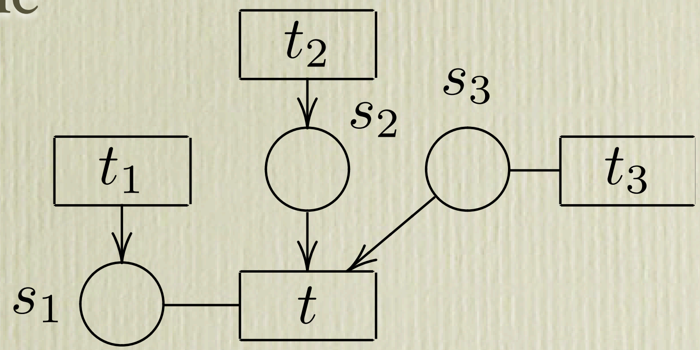
    take $\langle t, H_t \rangle$ with H_t of minimal size

    if $\langle t, H_t \rangle$ is a cut-off
        then do nothing
        else insert H_t (and possibly t) into Fin
            update PE

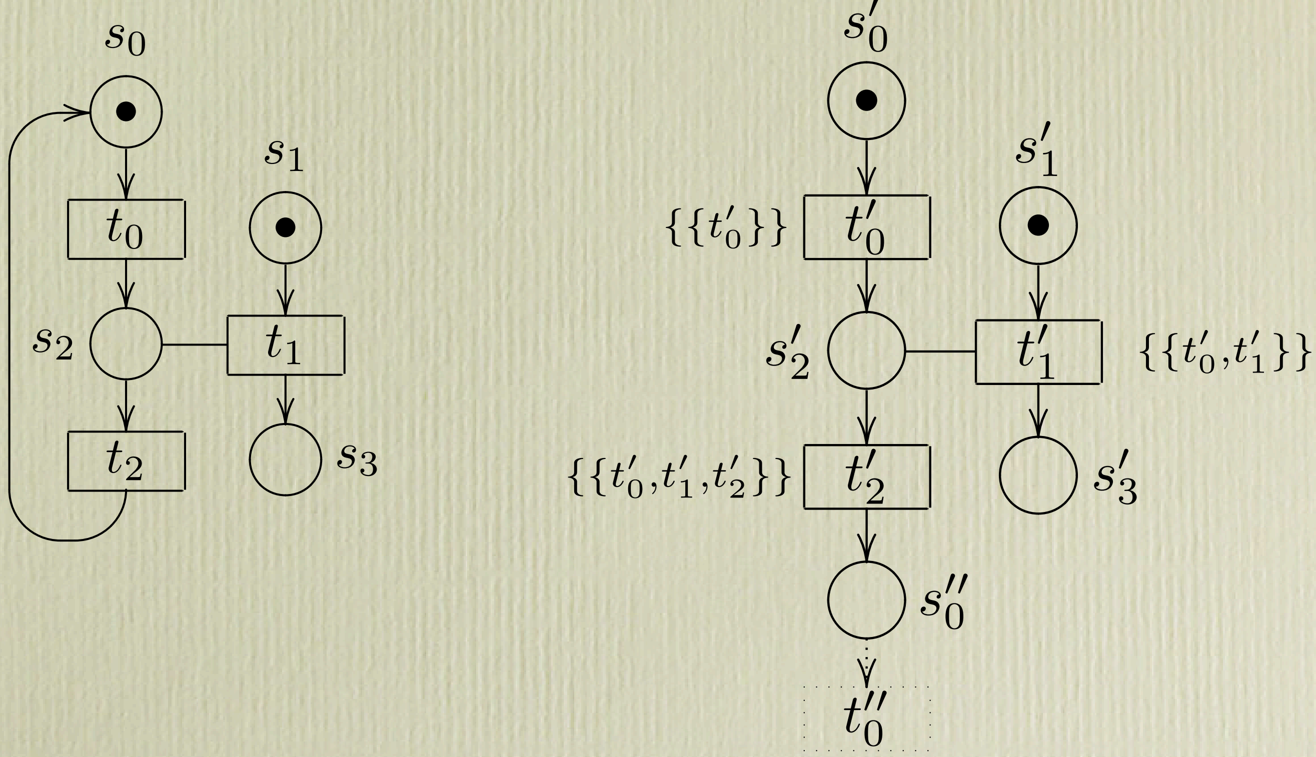
enddo

Updating PE (cont.)

- An history of t is obtained by taking
  - An history for each $t' \leq t$
  - Optionally an history for each $t' \nearrow t$
- ... pairwise compatible



Example



Results

- If the net is finite-state
  - The algorithm terminates
  - It produces a prefix whithout (local) cut-offs
  - The prefix includes the truncation hence it is complete

Compared to PR-encoding

- For safe nets
  - #histories = #transitions of PR-encoding
  - unfolding is a contextual net (smaller)
- For non-safe nets
  - #histories can be smaller #transitions of PR-encoding

Perspectives

- Implementation (including adequate orders)
  - suitable data structures for
    - representing histories (e.g., backpointers to events)
    - for updating PE (e.g., chains of asymmetric conflict)
  - experimental comparison with PR-encoding

Perspectives

- A general technique applicable to other formalisms
  - inhibitor nets
  - graph tranformation systems
  - RS in adhesive categories